

車載アプリケーション・ソフトウェアの移植性を高める通信仕様とその実装

——下位層を意識せずECU間のインターフェースを実現する「OSEK COM」

システムの記事

服部博行

ここでは，自動車のECU（electronic control unit）間の通信仕様「OSEK COM」について解説する．この仕様は，欧州の自動車メーカなどを中心とした車載制御ユニットの標準化に関するプロジェクト「OSEK/VDX」で定義されているものである．OSEK COM仕様に準拠した通信モジュールを用いることで，アプリケーション・ソフトウェアの開発者は物理層などの下位層を意識せずに，設計を行えるという．本稿では，まずOSEK COMの概要を説明する．また，筆者らが開発した本仕様に準拠した通信用ソフトウェア・モジュールについても紹介する．このモジュールは，車載用リアルタイムOS「TOPPERS/OSEKカーネル」に対応している．（編集部）

現在，自動車の利用者が要求する多くの機能は，ECU（electronic control unit）と呼ばれる電子制御装置（プロセッサと各種デバイスが組み合わさった一種のコンピュータ装置）で実現されています．自動車に搭載されるECUの数は年々増加し，より細密な制御を実現してきました．一方，自動車に求められる機能の多くは，一つのECUだけで完結するものはほとんどなく，複数のECUが協調して機能を実現しています（**図1**）．

このようにECUどうしが協調して，共通の機能を実現するためにはなんらかの通信手段を用いて情報を交換する必要があります（右掲のコラム「車載ネットワークの発展」を参照）．この情報交換のための通信を「車載ネットワーク」，「車載プロセッサ間通信」などと呼んでいます．現世代の車載ネットワークの代表例としては，CAN（controller area network）とそのサブネットであるLIN（local interconnect network）が挙げられます．一般に車載ネットワークという場合，物理層（トランシーバなど）について議論されることが多く，比較的多くの書籍で解説されています．

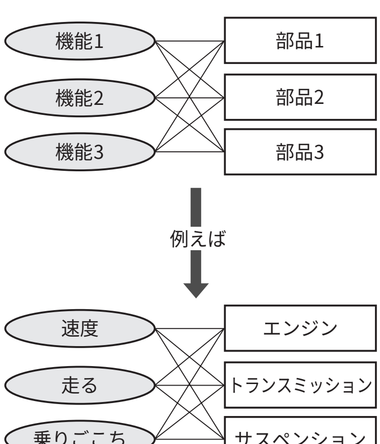


図1　自動車におけるECUの協調制御

自動車では，複数のECUが協調して一つの機能を実現することが多い．例えば，乗りごこちの良さは，エンジンやトランスミッション，サスペンションなど，多くの装置を用いて実現している．また，車両姿勢制御など高度な安全システムなどにおいても，ECUの連携による機能の実現はあたりまえとなっている．

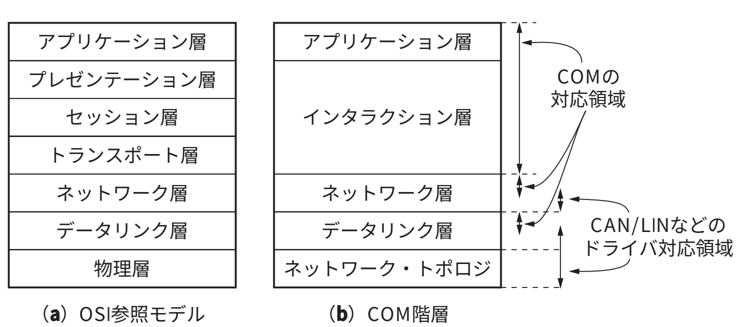


図2　OSI参照モデルとOSEK COM階層

OSEK COM階層では，インタラクション層にOSI参照モデルのプレゼンテーション層，セッション層，トランスポート層を含んでいるが，機能としてはOSIで規定している階層モデルを踏襲している．そのため，アプリケーション・ソフトウェアはアプリケーション層が提供しているAPI（application program interface）を用いることで通信機能が実現でき，移植性に優れる機能を兼ね備えているといえる．



コラム 車載ネットワークの発展

TOPPERSプロジェクトの会長である名古屋大学の高田広章氏は，自動車制御ネットワークの発展について，**表A-1**のような段階でとらえることができると発言されています．現在の車のステアリングを考えてみます．例えば，第2段階で実現されるステアリング補助機能では，操作する部位（ハンドル）と実際に操舵する部位（ステアリング・ギア）が物理的なパイプ（ステアリング・コラム）で接続されており，たとえ補助機能が障害により停止しても，最低限のステアリング機能は動作します．しかし，X-by-wireシステム（従来のメカニカルな機構を電子制御とネットワークで代替するシステム）になると，物理的な接続がないため，通信が停止した場合はステアリング機能が動作しないことになります．このように，通信機能は自動車の基本機能部位にまで浸透し始めています．

今後，車載ネットワークへの期待はさらに高くなることが予想されます．とくに，X-by-wireシステムなどは，ネットワークが自動車の基本機能を担うこととなり，高速な通信速度と高い信頼性が求められます．

表A-1　車載ネットワークの発展段階

段　階	内　容
第0段階	●自動車制御にネットワークが使われていない段階 ●エンジン制御，ブレーキ制御が個別に電子化（コンピュータ化）され，それぞれが独立に動作している段階
第1段階 （現世代ネットワーク：CAN通信など）	●各制御システムが車載ネットワークを通じて，制御品質を向上させるために有用なデータを交換する段階 ●各制御システムはおおよそ独立して動作 ●万一，ネットワーク障害が生じた場合，制御品質が下がりはしても，致命的な状況にはならない ●時間制約は比較的緩やか
第2段階 （現在，この段階の実用化が始まったところ）	●自動車制御ネットワークを通じて，ミッション・クリティカルでないサービスを提供する段階 ●個々の制御システムがネットワークを通じて協調動作することで新しいサービスを提供 ●ネットワークに障害が生じると，そのサービスは機能しなくなるものの，自動車の基本機能は保たれる．例えば，安定走行補助機能（トヨタ自動車のVDIM：Vehicle Dynamics Integrated Managementなど），車線維持システムなど
第3段階 （FlexRay通信はこの段階を目ざしている）	●自動車制御ネットワークを通じてミッション・クリティカルなサービスを提供する段階 ●ネットワークに障害が生じると，自動車の基本機能が損なわれる（X-by-wireシステムなど）

しかし，アプリケーション開発者から見た場合，かならずしも物理層を意識する必要性はなく，標準的なアプリケーション層が提供された通信仕様が規定されていれば問題ありません（**図2**）．本稿では，車載ネットワーク用のデバイスについてではなく，ネットワーク層からアプリケーション層を規定した通信仕様であるOSEK COM仕様について解説します．また，筆者ら（ヴィッツ）が開発した車載用リアルタイムOS「TOPPERS/OSEK カーネル」に対応したOSEK COMモジュールについても説明します．

1 OSEK COM仕様の概要

OSEK COM仕様は，欧州の自動車メーカを中心とした仕様策定団体「OSEK/VDX 注1」によって規定された，メッセージ・ベースの通信仕様です．そのため，OSEK COMはOSEK OSの利用を前提としていますが，OSEK COMの要求する各種サービス注2を満たしていればほかのOS（あるいはOSなし）の利用も可能です注3．

OSEK COM仕様の目的は，自動車制御装置向けソフトウェアの通信環境を提供することによって，通信インターフェースの共通化や内部通信および外部通信などの動作を規定し，移植性を向上させることです．

また，自動車制御装置に求められる機能は多様で，かつ利用できるリソースにも幅があり，ソフトウェアの規模もまちまちです．OSEK COMは，このような多様性にこたえるために，次に示す四つの適合クラスを用意しています（**表1**）．

- CCCA ――最小クラスで内部通信（Unqueued messagesによる送受信サービス）のみに対応する．メッセージを送受信できる
- CCCB ――内部通信（Unqueued message と Queued message による送受信サービス）のみに対応し，CCCAクラスのサービスおよび通信状態の参照を提供する

注1：「OSEK/VDX」は自動車のECU（電子制御ユニット）間のインターフェースに関する仕様であり，OS，通信，ネットワーク管理の三つについて定義している．
注2：タスク，イベント，割り込み処理サービスを必要とする．
注3：OSEK COMのCCCB（ECUの内部の通信を規定した適合クラス）は，部品として利用できるように非OS環境を意識した実装になっている．

- CCC0 ──内部通信と外部通信の両方に対応している．CCCAクラスのサービスに対応し，通知クラス2，バイト・オーダ（変換），ダイレクト送信モードをサポートする
- CCC1 ──最大の適合クラスであり，内部通信と外部通信に対応している．OSEK COMが規定するすべてのサービスを提供する

● 特定デバイスに依存しない通信を実現するCOM仕様

OSEK COMが提供するメッセージ通信サービスは，ECU内部通信とECU外部通信の2種類です．内部通信は，同一ECU内で稼働するタスク間，タスクと割り込みルーチンの間での情報交換を行います．外部通信は，ECU間をなんらかの通信デバイス（CAN/LINドライバ，シリアル・インターフェースなど）を経由して情報交換を行います．

図3に示すように，OSEK COMはインタラクション層，ネットワーク層，データリンク層から構成されていますが，ネットワーク層とデータリンク層の内部については規定していません．これは，ネットワーク層やデータリンク層は使用する通信デバイスなどに依存するためです．つまり，特定のデバイスを想定すると汎用性が損なわれるため，OSEK COM仕様では使用するデバイスを規定していませ

表1　OSEK COMの機能と適合クラスの対応

この表は，OSEK COMの機能と適合クラスごとの対応を示す．TOPPERS/OSEK対応のOSEK COMは適合クラスCCC1に対応している．つまり，OSEK COMの規定するすべてのサービスを提供できる．

出典：OSEK Communication Specification 3.0.2

機　能	内　容	CCCA	CCCB	CCC0	CCC1
Unqueued messages	受信メッセージに付与される属性でキューイングされないメッセージ．受信時にはつねに最新のメッセージに上書きされる．Zero-length messageとDynamic-length messageはUnqueued messagesの属性しか持てない	○	○	○	○
Notification class 1	通知クラス1．受信通知機能であり，メッセージを正常に受信したことを通知する．受信時にタスク起動，イベント設定，フラグ・セット，コールバックのいずれかを実行できる	○	○	○	○
Queued messages	受信メッセージに付与される属性でキューイングされるメッセージ．あらかじめ設定したキューイング数以上のメッセージは削除される．この場合，エラー通知は発生せず，その後の受信APIにてキューあふれがあったことを戻り値にて戻す．通常メッセージのみ設定できる		○		○
Message status information	メッセージ情報取得機能．Queued messagesのキューの情報を取得できる		○		○
External communication	外部通信機能．ECU間の通信機能を示す			○	○
Triggered transfer property	送信メッセージに付与される属性で，トリガ送信メッセージ．この属性が付加されたメッセージをSendMessageサービスで送信すると，ネットワークへのデータ送信を開始する			○	○
Notification class 2	通知クラス2．送信完了を通知する機能．メッセージが正常に送信を完了したことを通知する			○	○
Byte ordering	バイト・オーダ（変換機能）．COM仕様の送受信時にCPUやネットワークのデータ幅に対応したメッセージの変換を行う			○	○
Direct transmission mode	直接送信モード．周期的な送信を行わず，送信タイミングはユーザ・プログラムによって決定される．対応するのはPeriodic transmission modeとなる				○
Filtering	メッセージのフィルタリング機能．送受信時に指定されたルールでフィルタ処理を行う．変換条件が真ならばメッセージを通過させる				○
Pending transfer property	送信メッセージに付与される属性で，ペンディング送信メッセージ．この属性が付加されたメッセージをSendMessageで送信しても，メッセージがネットワークに流れず，メモリ上（I-PDU）に蓄えられる				○
Zero-length messages	メッセージ長がないメッセージ．すなわち，データがないメッセージであり，通知や送信トリガに利用できる				○
Dynamic-length message	可変長メッセージ．ただし，OILファイル（オブジェクトを定義したファイル）で最大長は決められている				○
Periodic transmission mode	定期送信モード．送信タイミングがあらかじめ決定される				○
Mixed transmission mode	直接送信モードと定期送信モードを組み合わせた送信動作を実現する				○
Minimum delay time	最短遅延時間．I-PDUを送信後，同一のI-PDUが送信可能になるまでの時間．Direct transmission modeとMixed transmission modeで有効				○
Deadline monitoring	指定された時間以内に送信を完了しない送信動作を監視する				○
Notification class 3	通知クラス3．受信エラー通知機能．受信時にエラーがあったことを通知する．外部からの受信時のみ有効.				○
Notification class 4	通知クラス4．送信エラー通知機能．送信時にエラーがあったことを通知する．外部への送信時のみ有効.				○
Callouts	フィルタリング，バイト・オーダ，UPDU（Underlying Layer PDU）格納時にいったんユーザ領域にメッセージを渡して，ユーザ独自のフィルタリングや，データ・パッキングを行う機能．OSのフック・ルーチンのような機能				○

ん(下掲のコラム「CAN通信との親和性が高い」を参照).

一方，インタラクション層はOSEK COMを利用するアプリケーション・ソフトウェアにOSEK COM仕様のサービス用APIを提供します． そのサービスは，メッセージ送信および受信サービスです．インタラクション層は，これより下位層のサービス提供(OSEK COM 仕様が規定していない各種デバイス)を受けて送受信を行います．そのため，OSEK COM仕様ではインタラクション層とネットワーク層やデータリンク層を接続するインターフェースを規定しています．

内部通信は，前述したように同一ECU内のタスク間通信であり，ECU間の通信を必要としません．そのため，データは**図3**に示されているネットワーク層やデータリンク層を経由する必要はなく，インタラクション層の内部で処理が完結します．すなわち，データ送信が指令されたメッ

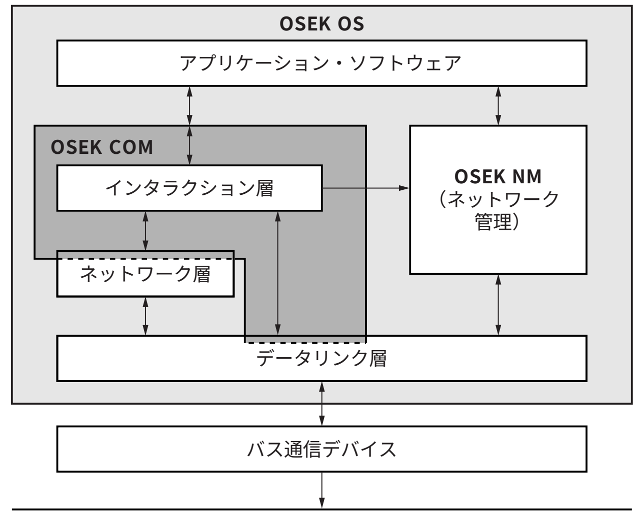


図3　OSEK COMの構成図 (1)

OSEK COMは，バス通信デバイスとアプリケーション・ソフトウェアの中間に位置する．アプリケーション・ソフトウェアは，OSEK COMとデータの送受信を行うため，通信先(バス通信)を意識しない設計が可能となる．

コラム　CAN通信との親和性が高い

本文では，OSEK COMの外部通信において通信デバイスは特定しないと述べました．しかし，実はCOMはCAN通信と親和性が高い仕様です．もともとOSEK COMの仕様書にはCAN通信を想定していることが明記されています．とくに，以下の三つの特徴から，CAN以外の通信はあまり想定していないと考えられます．

1) フィルタリング処理

フィルタリング処理は，通信タイミングに密接にかかわる機能です．フィルタリング処理は，送信データが変化したタイミングをとらえるために利用されると思われます．すなわち，データが変化するタイミングでデータの送信を行うという場合に利用される機能と言えます．このようなイベントによるデータ送信は，CAN通信とは親和性が高いと思われます注A-1.

一方，LINの場合，たとえデータが変化したというイベントを認識したとしても，送信タイミングはマスタ・ノードECUが管理しているため，すべてのメッセージは周期的に送信されます注A-2．つまり，イベントによる送信は考えられません．また，次世代の車載ネットワークとして注目されているFlexRayはネットワーク全体で各メッセージの送信スケジュールが規定されている方式であるため，OSEK COMを利用するメリットは大きくありません．

ただし，LINやFlexRayでも受信時にフィルタリング処理を利用することは考えられます(例えば，受信したデータが変化しているかどうかで処理実行の有無を判断する).

2) Dynamic-length message(可変長メッセージ)

Dynamic-length messageは，メッセージ長の最大長は規定されているものの，それ以下のメッセージであれば動的に変更できる機能です．CANは，この機能をサポートしているため利用可能です．しかし，LINは静的にメッセージ長が決定されているため，利用することができません．

3) 通知機能

CAN通信のデータ送信は静的に送信スケジュールが決定されていないため，データ受信時などは通知機能を用いるほうがアプリケーションの開発には効率的であると考えられます．

しかし，LINなどに代表される周期送信方式の通信では，かならずしも通知機能は必要なく，ポーリングによる受信で問題ありません．

このように，OSEK COM仕様の下位層(UL：Underlying Layer)がまさにCAN通信の特徴をそのまま要求しているため，CAN通信との親和性が高いと考えます．

すなわち，CANはマルチマスタ方式(どのノードでもデータ送信を開始できる)で，データはECUごとの時間(タイミング)で送信します．このような特徴がそのままCOM仕様に結びついています．

シングルマスタ方式のLINでは，ノードみずからがなんらかのイベントでデータを送信することは想定されていません．また，FlexRayはネットワーク上に時間情報を保有しています．この考えはOSEK COMに含まれていません．そのため，OSEK COMではFlexRayへの対応が難しく，OSEK/VDXが策定する「FT-COM(タイム・トリガのアクセス方式をとり，耐故障性を備えた仕様)」に類似したCOM仕様が必要となります．

注A-1：CANのアクセス方式はイベント・ドリブン(事象駆動)である．なんらかのイベントの発生によってメッセージ送信を行う．

注A-2：LINのネットワーク・トポロジは，一つのマスタ・ノードと複数のスレーブ・ノードからなる．あらかじめ定められたスケジュールでマスタ・ノードがスレーブ・ノードと通信を行う．つまり，周期送信のみを扱うことができる．

セージは，インタラクション層内部で受信データ・フォーマットに変換して，あたかも受信があったようにアプリケーション・ソフトウェアに渡すことができます．

図4は，インタラクション層の内部構造を示した図です．インタラクション層は内部で二つの層に分かれています．上層部はAPIを提供する部位で，ここで内部通信を実現します．下層部はインタラクション層の下位層と接続して下位層への処理要求や情報を受信する部位で，ここは外部通信時のみ利用されます．**図4**から理解できるように，インタラクション層が提供するAPIは，内部通信と外部通信で共通です．つまり，OSEK COMを利用すれば，アプリケーション・ソフトウェアは通信相手が内部のタスクなのか，外部ECUに含まれているオブジェクトなのかを区別することなく，メッセージ通信できるというメリットがあります．

● 通信処理とネットワーク管理をOSから独立させた理由

OSEK COMは，本来ならOSに含まれる機能だと考えられます．例えば，μITRONでもOSの機能の一部としてOSEK COM仕様が定めている内部通信を規定しています．では，なぜOSEK/VDXでは，OSEK OSとOSEK COMを分けて定義しているのでしょうか．

車載ECUのもっとも小さい製品カテゴリでは，かならずしも内部通信機能を必要としていないケースがあります．理由はいくつか考えられますが，小さいECUほどプロセッサの能力が低く，タスク間通信による情報交換を行うと情報のやり取りに伴う手続きのオーバヘッドが無視できなくなるからです．反対に，比較的大きいECUはほかのECUとの情報交換が必要となり，外部通信機能が必須となります．このような前提から，OSEK/VDXではOS仕様に特定の通信を対象とした仕様を盛り込むのではなく，通信仕様のみを規定したOSEK COM仕様としてOS仕様から独立させたものと考えられます．

同じように，外部通信で用いられるネットワーク管理機能もOSEK NM（network management）仕様として規定されています．ネットワーク管理機能は車両内で統一したネットワークの送受信の手順を確立するものですが，一般にこうした機能は自動車メーカごとに独自の仕様が規定されています．そのため，ネットワーク管理機能も基本的な仕様のみを定義し，OS仕様から独立させています．

図5に，OSEK/VDXで規定されている仕様の構成を示します．この図からわかるように，OSや各種デバイスのサービスを利用しながら，OSEK COMが通信処理をアプリケーション・ソフトウェアに提供します．すなわち，アプリケーション・ソフトウェアからは，通信指令先はつね

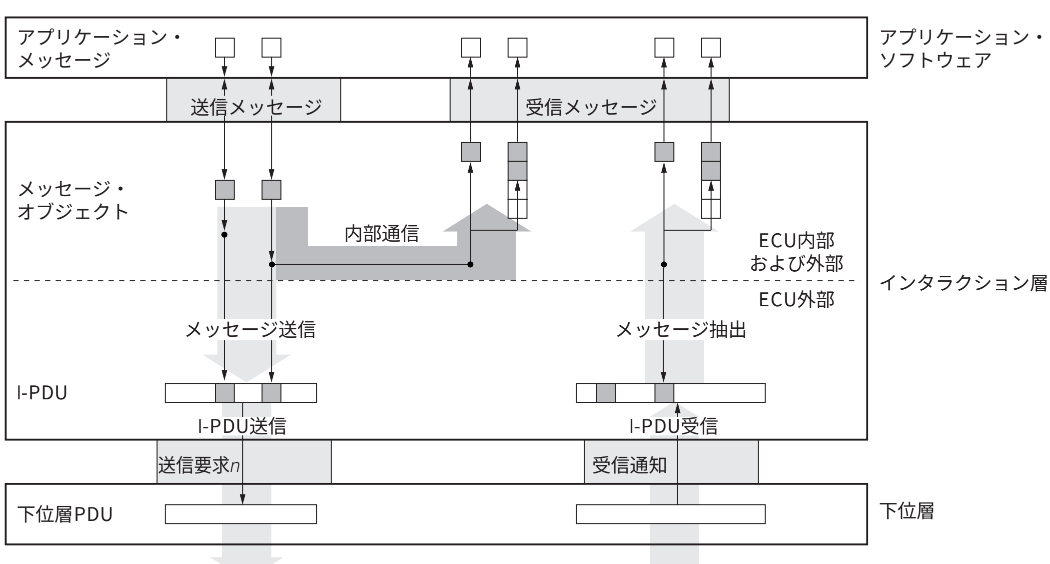


図4　OSEK COMのメッセージ送受信モデル (1)

OSEK COMの送受信モデルを示す．アプリケーション・ソフトウェアは送信メッセージAPIを用いてCOMにメッセージを送信する．COMはそのメッセージが内部通信であるか，外部通信であるかを判断し，内部通信の場合はそのまま受信メッセージAPIでアプリケーション・ソフトウェアにメッセージを送る．外部通信の場合，COMの下位層を経由してほかのECUにメッセージが送信される．COMの下位層で受信したメッセージはCOMを経由して，受信メッセージAPIにてアプリケーション・ソフトウェアにメッセージが送られる．このように，アプリケーション・ソフトウェアは受信したメッセージが内部通信，外部通信にかかわらず同一のAPIを用いて受信できる．

にCOMであり，デバイスを意識する必要はありません．また，COMとNMが協調してネットワーク管理を制御するため，ネットワークの接続状態参照やネットワークの起動，状態変更などを行わないかぎり，アプリケーション・ソフトウェアから直接NMへの処理を指令する必要はありません．

なお，NMはCOMとは密接に関係するモジュールですが，仕様上はむしろ通信デバイスと深く関係します．例えば，CAN通信を実現する場合，単に送受信を実行するだけであれば，CANドライバとCOMが接続していることで実現できます．しかし，この機能だけではCANの特徴であるマルチマスタ方式を実現できません．マルチマスタ方式では，CANネットワークに接続された各ノードECUがバス状態をつねに監視し，ネットワーク接続状況を更新して，ECUの脱着やネットワーク構成の変更が行われても正しく通信を継続します．また，車両状況により，ECUが動作する必要がない場合，通信バスを休止状態にして消費電力を抑えるなどの機能（ウェイクアップ/スリープ機能）が必要になりますが，これもネットワーク管理の重要な機能の一つです．

このように，NMは通信デバイスと密接に関連します．また，CANとLINなど，同一システムに複数の通信デバイスを利用するゲートウェイ装置などの場合，COMモジュールは一つでも，NMモジュールはCAN用，LIN用のそれぞれが必要となります（p.131のコラム「通信デバイスが同じならNMモジュールは共通？」を参照）．

● **COM仕様のねらいはソフト部品の移植性を高めること**

上述したように，OSEK COMは内部通信，外部通信を含めた通信機能を一つの仕様として規定するため，アプリケーション設計者が通信相手との接続方法を意識する必要がありません．このことは，単に設計者が必要とする各種の調査を含めた設計期間を短縮するだけでなく，アプリケーションをほかの環境に移植する場合や接続方法を変更した場合の修正を低減させる効果もあります．

図6（a）に示すように，COMがない場合はデータの送受信を行う際にタスク設計者がすべてを管理する必要があります．つまり，ほかのタスクと情報を交換したい場合，あらかじめ決めておいた手段に従い，情報を交換します．また，ほかのECUと情報を交換する場合も，そのECUとの

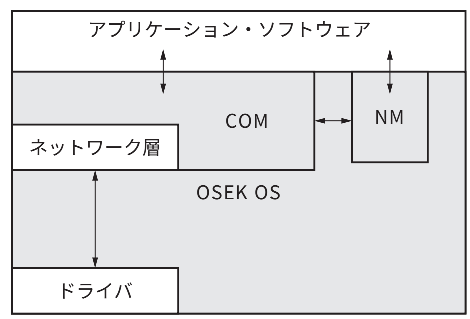
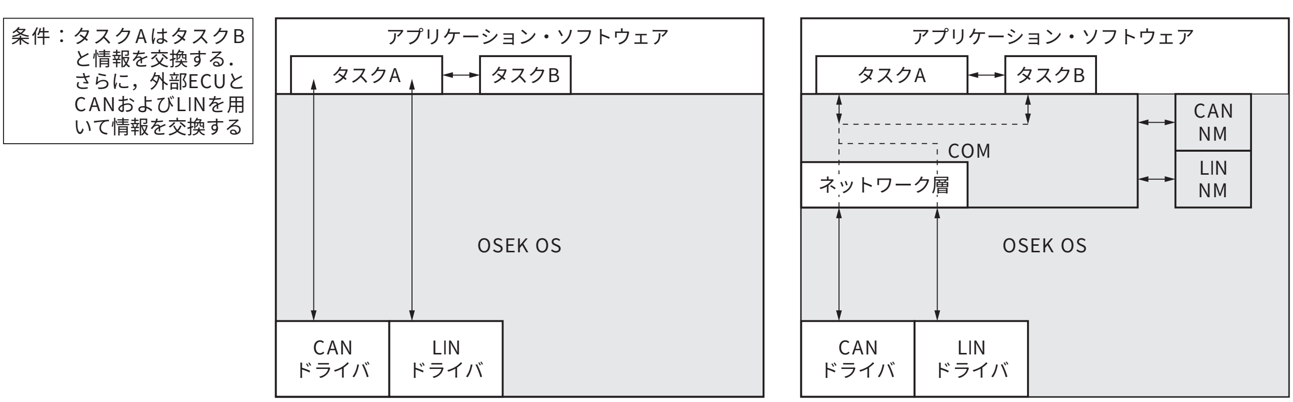


図5　OSEK/VDXにおけるOS，通信仕様，ネットワーク管理機能
COM（通信仕様）はOSEK OS上で稼働し，NM（ネットワーク管理機能）と協調してネットワーク通信を実現する．アプリケーション・ソフトウェアから単にメッセージ通信を行う場合はCOMを介して送受信が行われる．ただし，ほかのECUがネットワークに参加，離脱するといった場合を的確に捕捉するために，アプリケーション・ソフトウェアはNMの機能を直接利用して，ネットワークの状態を管理する．



（**a**）COMなし　　（**b**）COMあり

図6　COMの有無によるアプリケーション開発の違い
（a）のCOMなしの場合，タスクAが直接CAN/LINドライバを操作し，タスクBともなんらかの手段で情報を交換する．つまり，タスクAは，CANドライバ，LINドライバ，タスクBとの間で異なる情報交換手段を用いるため，個別の設計が必要となる．一方，（b）の場合，タスクAはCANドライバ，LINドライバ，タスクBとの通信はすべてCOMを経由する．COMには各種情報の通信手段が静的に指定されているため，識別情報で接続先を自動的に変更し，情報交換手段の差異をCOM内部で隠ぺいする．そのため，通信先を意識してタスクAを設計する必要はなく，それぞれに対して同一APIによる通信が可能となる．

接続状況を把握して，タスク設計者が個別に情報交換処理を設計する必要があります．このように，あらゆる情報交換において設計者がすべてのシステム構成を把握しておく必要があり，システム構成が変更された場合などは再設計しなければなりません．このことは，タスクの部品化を阻害するものであり，どんなにほかのタスクと疎結合（独立）になるような設計を行っても，通信相手との情報交換手段が密な開発であれば，単純に移植などできません．

一方，COMを利用する場合，タスクの設計者は通信相手との接続情報を知る必要がなく，通信したい相手の識別番号を用いて通信するだけで，COMが自動的に通信手段を決定して送受信を行います（**図6(b)**）．すなわち，通信相手が同一ECUに含まれる内部通信による情報交換相手と，ほかのECUに含まれる外部通信による情報交換相手

リスト1　ソフトウェア記述の比較

COMを利用すると，すべてが送信メッセージAPI（SendMessage）で送信できる．接続先はCOMが自動認識し，メッセージ指定の接続先にデータを転送する．また，アプリケーションでデータ更新管理や排他制御を行う必要がない．接続先をCOMが判定するため，メッセージの接続先が変更されてもアプリケーションを変更する必要がない．

```
// CAN周期送信メッセージの送信
SendCanMessage ( &pszCanCyvleMsg );
//CANイベント送信（送信データに変更があれば送信）
If(isChangeMessage(&pszCanEventMsg,&pszCanLastEventMsg);
    SendCanMessage ( &pszCanEventMsg);
}
// LIN通信によるメッセージ送信
SendLinMessage( &pszLinMsg);
 // タスクBとの通信（変数による交換）
if( isAccess(TASK_B)){
    nTaskB_Var1 = Var1;
    nTaskB_Var2 = Var2;
}
```

（**a**）COMなし

```
// CAN周期送信メッセージの送信
SendMessage ( &pszCanCyvleMsg );
//CANイベント送信（送信データに変更があれば送信）
SendMessage ( &pszCanEventMsg);
// LIN通信によるメッセージ送信
SendMessage( &pszLinMsg);
 // タスクBとの通信（変数による交換）
SendMessage (&pszTaskB);
```

（**b**）COMあり

※このプログラムはあくまでも概念的なことを示しているだけであり，実際の製品を想定した記述ではない．

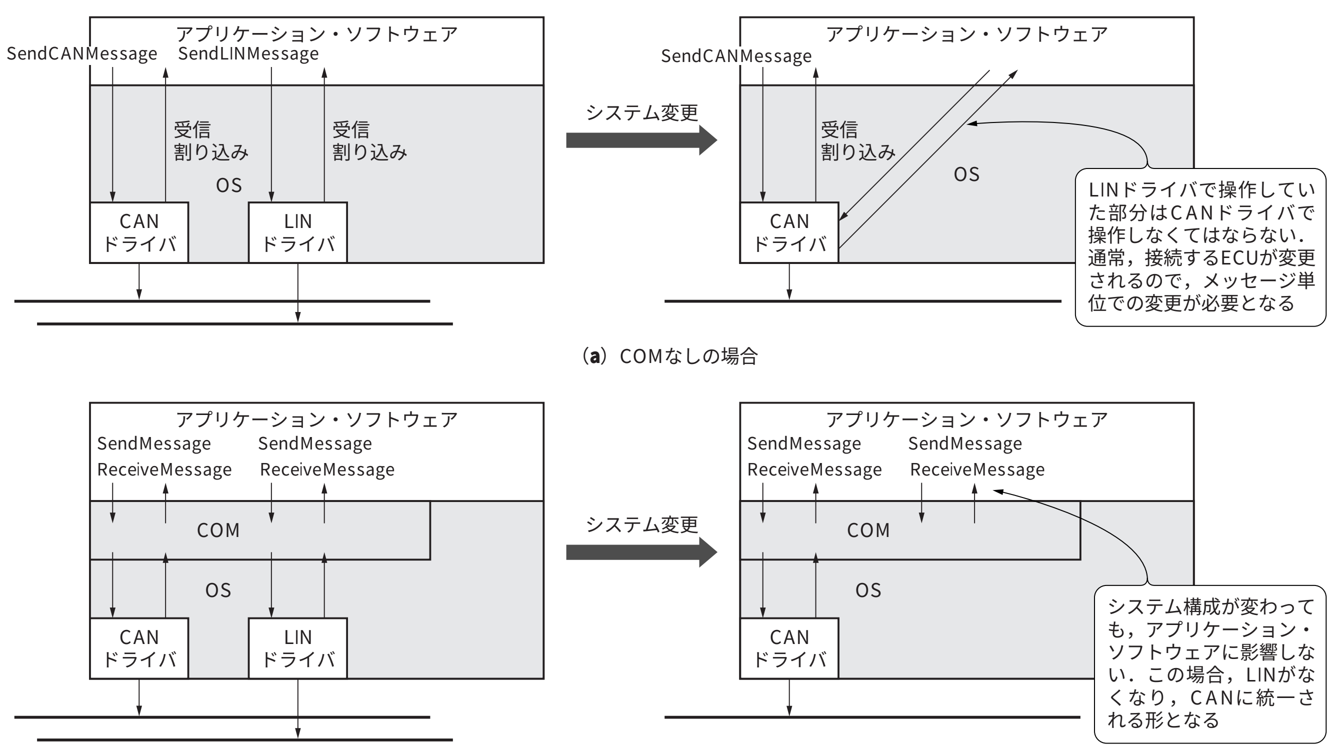


図7　COMの優位性

設計後，システム構成の変更によってメッセージの接続先が変わった場合，COMを利用していればアプリケーション・ソフトウェアに影響を与えない．

を区別する必要がありません．このように同一APIによる情報交換を可能とすることによって，タスク単位の移植性の向上はもとより，接続などのシステム構成を設計終了後の性能調整の段階で変更することも可能となります（**リスト1**，**図7**）．

このように，COMを用いることによってタスク単位の独立性を確保し，移植やシステム変更などが容易に実現できる環境を構築できます．

また，COMのフィルタリング処理を利用すれば，メッセージ送受信のタイミングを任意に制御できます．例えば，データの変更があったときにのみある情報を送信したい場合，COMのフィルタリング処理でデータ変更をとらえて送信を行います．すなわち，COMの機能でデータ変更を認識するため，アプリケーションで前回の送信データと比較する必要がありません．

現在の自動車通信ではすでにCANの帯域幅を使いきってしまい，すべての情報を送信できない状況になりつつあります．そのため，値が変更したときのみ送信を行い，通信の負荷を低減させたいという要求が増えています．COMを利用しない場合，このような送信はアプリケーションが管理することになります．

2 TOPPERS/OSEKカーネル対応のOSEK COM

本来，OSEK COM仕様は，OSEK OSの環境下で動作することを想定して策定されています．しかし，アプリケーション開発者はかならずしもOSとCOMの両方の機能を求めているわけではなく，どちらか一方のみを利用したい（あるいはどちらか一方を選択せざるをえない）というケースもあります．

このような状況の場合，車両制御以外のソフトウェア開発ではOSを使用せず汎用の通信モジュールを利用するというよりも，OSを利用して専用の通信モジュールを新たに開発するというケースが多いように筆者は感じます．

しかし，車両制御ソフトウェアの開発では，かならずしもOSのほうが通信モジュールよりも多く選択されるとは断定できません．事実，自動車ではOSを利用しないシステムが大半を占めているわりには注4，通信モジュールであるCOMを使用したいというケースが多くあります．これは，ECUのハードウェア・リソース条件が厳しく，OSと通信モジュールの両方の利用が難しいことと，ECU間の通信機能が必須であるためだと考えられます．

1台の車には非常に多くのECUが搭載されて高度な機能を実現しています．しかし，それらのECUは異なるメーカから提供されたものであり，たとえ同じ通信規格を用いていても，通信異常時の処理などの細部の仕様は各社各様であり，ECU間の接続の検証は困難を極めています．そのため，厳密な通信規則を守って通信処理を実現するには，統一された仕様が必須となります．このような疎結合のマルチプロセッサ・システム注5には，通信処理はある意味，制御の要であると言えます．そのため，車両制御ソフトウェアにおいてOSを利用せずにCOMを利用するケースがあると考えられます．

このような理由から，TOPPERS/OSEKカーネル対応のOSEK COMの実装は，基本的にOSEK OS環境下で動作しますが，OSを利用しない場合の実装も視野に入れています．

注4：ボディ系ECUの多くはOSを使っていない．制御系ECUにおいても，比較的大きな制御システムを除いて，現状はOSを使用していない．
注5：プロセッサごとにOSが存在し，複数のプロセッサが独立して動作するシステム．

コラム 通信デバイスが同じならNMモジュールは共通？

通信デバイスが異なれば，ネットワークの管理方法が異なるのも理解できます．では，通信デバイスが同じなら，ネットワーク管理を共通化して利用できるのでしょうか？

これは理論的には可能ですが，実際は車両内で統一した管理方法でなければなりません．残念ながら，ネットワークの管理方法は自動車メーカ独自のノウハウに基づく障害対策を施して通信の信頼性を向上しているため，自動車メーカ間の互換性はありません．

結果として，ネットワーク管理は，自動車メーカごとの独自仕様となり，共通化が難しい部位であり，NMモジュールの共通化は難しいといわざるをえません．

このような自動車メーカごとに異なる仕様は，自動車メーカをまたがった電装部品の流用を阻害します．また，接続検証などにも時間を要するため，国内の自動車メーカおよび電装部品メーカが共通化を模索して「JasPar（http://www.jaspar.jp/）」と呼ばれる団体を設立しました．

● データ伝送速度を落とさない内部通信の実装

内部通信機能は，タスク間のメッセージ通信を実現する機能であり，OSの基本機能として位置づけても問題ないと言えます．また，外部通信を必要としないシステムでも，COMを利用することでタスク間の結合を抑えること（疎にすること）が可能です．

しかし，COMを用いたメッセージ通信を利用してタスク間のすべてのデータ交換を行う場合，グローバル変数を用いたデータ交換と比較すると，データの伝送速度の低下が著しいという評価結果が出ています[注6]．そのため，TOPPERS/OSEKカーネル対応のOSEK COMは，オーバヘッドを極力少なくするために，内部通信は実質OSEK COMモジュール内に用意したグローバル変数領域を用いて情報交換を行い，実行効率の低下をできるだけ損なわない実装としています．

また，OSEK COM仕様を拡張した機能として，ロギング情報[注7]の取得を容易にするため，ログ取得マクロを実装しています．基本的には，OSEK COM仕様で規定されているすべての内部通信機能（適合クラスはCCCB）を実装しています．

● 外部通信が現在対応しているのはCANとLIN

外部通信機能は，データ受信用のタスクをCOM内部で

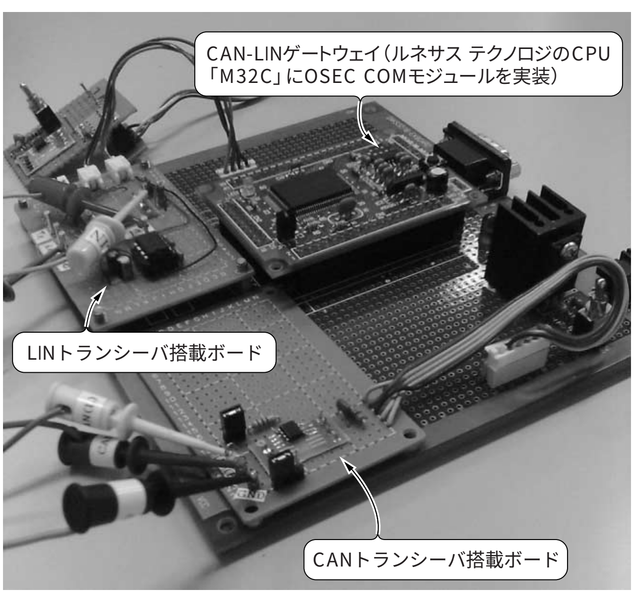


写真1　TOPPERS/OSEKカーネル対応OSEK COMモジュール

筆者らが開発したTOPPERS/OSEKカーネル対応OSEK COMモジュールを実装したボード．CANとLINのデバイス・ドライバも実装されており，CAN通信なども行える．

静的に生成しています．そのため，基本的にOSでの利用を想定していますが，受信用タスクに類似する機能を用意すればOSを使わない環境下での利用も可能となります．また，時間管理用モジュールとしてOSのアラーム機能に類似した機能を実装しています．これも，OSを利用しなくてもCOMモジュールのみでCOM仕様が実現するように考慮している点です．また，内部通信と同じように，外部通信もログ取得マクロを実装し，ロギング情報の取得を容易にしています．

機能的な面では，OSEK COM仕様のすべての外部通信機能（適合クラスはCCC1）を実装しています．

現在，対応している外部通信デバイスは，CANドライバとLINドライバです（**写真1**）．これらのデバイス・ドライバはいずれも筆者らが開発しました．TOPPERS/OSEKカーネル対応OSEK COMモジュールを公開する際には，これらのデバイス・ドライバも公開する必要があると考えています．そのため，デバイス・ドライバについての品質検証を実施する予定です．

また，今後はシリアル・インターフェースやEthernetなどのデバイスへの対応も予定しています．さらに，FlexRayへの対応も考えています（ただし，現在のところまだFlexRayそのものの仕様が固まっていないので，対応は仕様確定後となる）．

本OSEK COMモジュールの内部エラー・コードは，独自の拡張を施したものを利用していますが，利用の際にとくに注意するべき項目ではありません．

なお，ここで紹介したCOMモジュールは，2005年度の経済産業省「地域新生コンソーシアム事業」として採択された「自動車統合制御用組込みOSの開発」の基幹部品の一つです．この事業において，本COMモジュールを改良し，信頼性を検証した後，プロジェクト会員向けに早期リリースを行う予定です．公開スケジュールは，2005年11月に開催される「Embedded Technology 2005」で紹介できると思います．

● 車載ソフトウェア開発のプラットホームを目ざす

TOPPERS/OSEKカーネルが目ざすのは，自動車制御プ

注6：自動車メーカは，グローバル変数による情報交換とCOMのメッセージ通信を用いた情報交換の効率の比較を行っている．
注7：例えば，COMの開発（検証）を行っているときは，COMのエラー状況などを出力して，APIのコール順序やエラー発生時のトレースなどに利用する．また，アプリケーション開発時には，アプリケーションから要求されたメッセージ送受信の実施ログなどを取得できる．

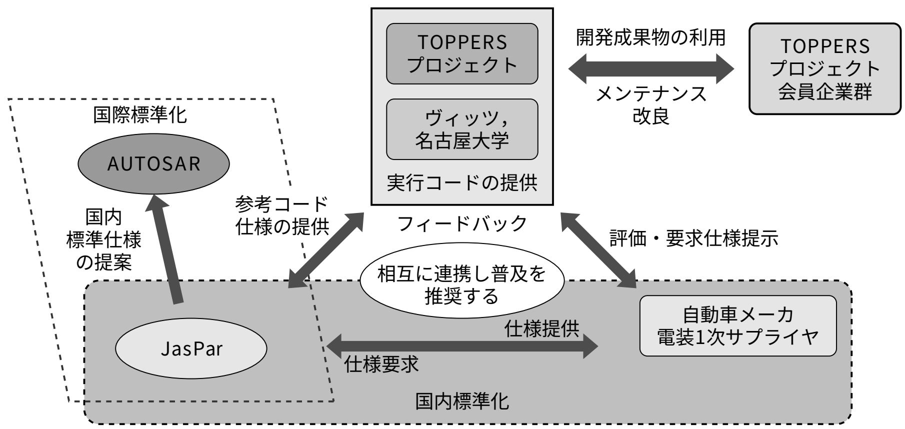


図8
TOPPERS/OSEKの目ざすもの
筆者らと名古屋大学で，積極的にソフトウェア部品を含めたリファレンス・デザインの開発を行い，この開発成果をTOPPERSプロジェクトにコントリビュートする．さらにJasPar（Japan Automotive Software Platform Architecture）に提案して同業界団体のリファレンスOSとなり（現時点では改善すべき点が多々ある），最終的には世界標準を目ざす．図で具体的に説明すると，最下部の四角はJasParを含み，国内の標準化を目ざすものである．そして，左側の台形はJasParとAUTOSARを含み，国際標準化を目ざしている．

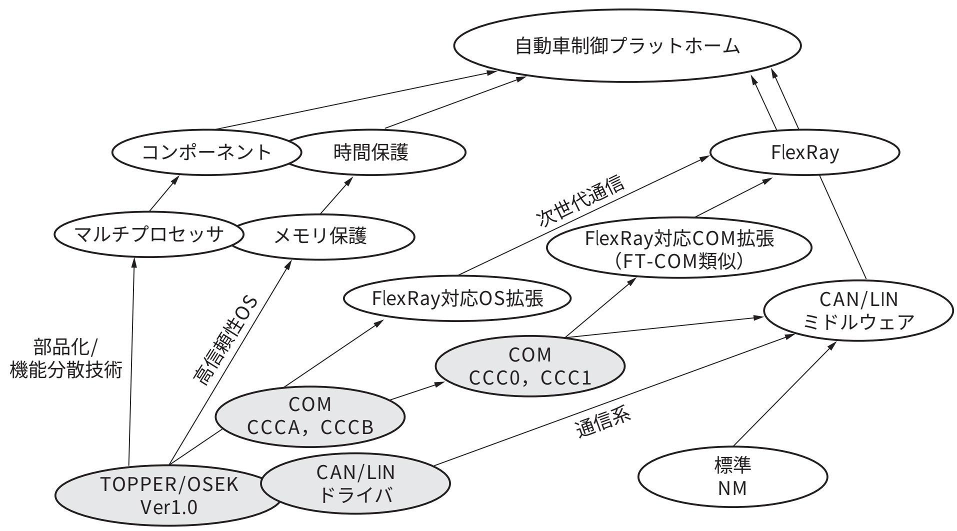


図9
TOPPERS/OSEKのロードマップ
今後の開発予定を示す．灰色で示した部分はすでに開発済みのもの．

ラットホームのリファレンス実装です（**図8**）．そのため，現世代の要求事項はもちろんのこと，次世代の車両開発に必要となると思われる機能を積極的に開発していきたいと考えています．今後は，以下に示すようなカーネルや機能の開発を行う予定です（**図9**）．

- 次世代カーネル——次世代カーネルは，メモリ保護機能，時間保護機能，マルチプロセッサ機能，コンポーネント機能から構成される
- 通信機能——現世代通信であるCANやLINに対応した通信ミドルウェアを開発する予定．また，次世代通信としてはFlexRayへの対応を検討中．FlexRayへ対応するためには，OSEK/VDXが定める「Time-Triggered OS（OSEKtime）」に類似したOSの開発，FT-COMに類似したCOMモジュールの開発が必要
- 用途別カーネル——自動車には多種多様な機能が要求されており，それらを実現するためには特定機能に特化したカーネルを必要とする場合が多く想定される．それら特定用途向けのカーネルを開発する予定

参考・引用*文献

(1)* OSEK/VDX；Communication Version 3.0.3，July 20 2004.
(2) OSEK/VDX；Operating System Version 2.2.1 OSEK/VDX, January 16 2004.
(3) OSEK/VDXのWebサイト，http://www.osek-vdx.org/.
(4) 高田広章；オープンソースのOSEK仕様OS，Automotive Technology Days 2004 Autum，2004年11月.

はっとり・ひろゆき
(株)ヴィッツ

<筆者プロフィール>
服部博行．自分の興味のあるものは自分で作りたがるやっかいな性格．日曜大工，ビール作りなどで自分の時間を楽しんでいる．最近は，2人乗りの車でドライブを楽しみ，家族からは白い目で見られているが，本人はまったく気にしていない．